

# GOP 기반의 적응적 인터 예측을 이용한 다시점 비디오의 효율적인 H.264/AVC 부호화

정회원 이 정 호\*, 조 익 환\*, 이 응 호\*\*, 정 동 석\*\*\*

## An Efficient H.264/AVC Encoding Using GOP Based Adaptive Inter Prediction

Jungho Lee\*, Ikhwan Cho\*, Woongho Lee\*, Dongseok Jeong\*\*\* *Regular Members*

### 요 약

본 논문에서는 H.264/AVC의 다시점 비디오 부호화의 고속화 방법을 제안한다. 다시점 비디오는 부호화해야 할 데이터의 양이 일반 비디오보다 더 많으며, 전형적인 비디오 부호화 방법에서 사용하는 인터-움직임 예측에 덧붙여 인터-부등 예측을 이용하기 때문에 예측해야 할 정보가 막대하게 커진다. 실험을 통해 비디오 부호화에서 사용되는 각 예측의 효율이 인트라, 인터-부등, 인터-움직임 그리고 인터-skip 순으로 점점 좋아진다는 점에 착안하여 각 GOP 단위로 적응적인 문턱값을 계산하여 인터-부등 및 인트라 예측을 조기 종료하는 고속화 알고리즘을 제안하였다. 실험 결과 제안하는 방법은 여러 실험 비디오에 대해서 기존의 방법에 비해 약 32%가량 개선된 연산 속도를 보였으며, 비트량 및 왜곡의 증가는 상대적으로 적었다.

**Key Words** : Multi-view, MVC, Inter Prediction

### ABSTRACT

This paper proposes a fast encoding algorithm of H.264/AVC multi-view video coding. The amount of data to be encoded for a multi-view video is much more than normal video's data, and the amount of information to be predicted is enormous because of the multi-view video coding uses inter-disparity prediction in addition to inter-motion prediction in conventional video coding. We noticed through an experiment that the efficiency of prediction is getting better in order of intra, inter-disparity, inter-motion, and inter-skip, and proposes a early termination algorithm by means of estimate the adaptive threshold within a GOP unit. In the experiments, the proposed algorithm shows improved processing speed about 32% compared to existing method, and increased amount of bits and distortions are relatively disregarable.

### 1. 서 론

최근 IT기술의 눈부신 발달로 인하여 멀티미디어 서비스에 대한 수요가 나날이 증가하고 있는 시점에서 그 수요에 부합하는 응용 시스템의 개발 가운데 멀티미디어 데이터 압축의 중요성은 아무리 강

조해도 지나침이 없다. 멀티미디어 데이터 압축 방법은 MPEG, MPEG-2, MPEG-4, 그리고 H.261, H.263 등으로 발전되어 왔고, 최근에는 H.264/AVC 라는 표준이 아직까지 연구 중에 있다. 이런 대부분의 압축 방법들은 하나의 멀티미디어 데이터에 하나의 영상을 부호화하는 방향으로 진행되어 왔는데,

\* 인하대학교 전자공학과 멀티미디어 연구실 ({julian, teddydino}@inhaian.net), \*\* ltlee@lge.com, \*\*\* dsjeong@inha.ac.kr  
논문번호 : KICS2006-10-433, 접수일자 : 2006년 10월 16일, 최종논문접수일자 : 2006년 12월 7일



그림 1. 다시점 비디오의 예. 위에서부터 ballroom, race1, uli  
Fig 1. Examples of multi-view video sequences. From top most ballroom, race1, and uli

H.264/AVC의 최근 연구는 하나의 멀티미디어 데이터에 여러 다른 시점에서 촬영한 영상을 부호화하여 담는 이른바 멀티-뷰 비디오 부호화(Multi-view Video Coding, MVC)까지 그 연구 범위를 확장시켰다<sup>[1]</sup>.

MVC는 사용자가 원하는 시점의 화면을 골라서 시청할 수 있는 자유 시점 비디오(free view point video, FVV), 3차원 디스플레이 장비를 이용한 3차원 비디오, 그리고 여러 명이 참여할 수 있는 화상회의 등의 여러 가지 새로운 응용분야에 사용될 수 있으며, 응용 방법에 따라 수많은 다른 분야로도 확장될 수 있다<sup>[1][2]</sup>.

이렇게 많은 이점을 가지고 있음에도 불구하고, 동시에 여러 개의 비디오를 부호화해야 하기 때문에 MVC는 결과물의 비트량이 일반 비디오보다 월등히 크고, 또한 부호화하는데 필요한 계산량이 막대하다는 문제가 발생한다. 따라서 압축률을 개선하면서 계산량을 줄이는 것이 MVC에서 가장 중요하다고 말할 수 있다. 압축률을 개선하기 위한 방법은 여러 가지가 제안되었는데, 그 중에 체계적 B-픽처(hierarchical B-picture)를 이용한 방법과 사다리형 피라미드 GOP(lattice-like pyramid GOP) 방법은 상당히 좋은 성능을 보여준다. 그 외에도 키프레임 공유(shared key frame)를 이용한 방법, 카메라 간의 영상 차이를 보정하는 방법 등이 광범위하게 연구되면서 MVC의 비트율 및 화질을 개선시킬 수 있는 방법들이 제안되었다. 본 논문은 이들과는 시각을 달리하여 화질과 비트량은 여타의 방법과 비슷한 수준으로 유지하면서, 계산량을 줄여 부호화 속도를 향상 시키는 방법을 제안한다.

본 논문에서 제안하는 알고리즘은 두 가지 인데, GOP 단위로 첫 번째 B 슬라이스에서 매크로 블록의 정보를 수집하여 이를 다른 B 슬라이스의 부호

화에 이용하는 방법과 경우에 따라 인트라 예측을 생각하는 방법이며, 이 두 가지를 혼합하여 이용한다.

논문의 구성은 다음과 같다. 2장에서 MVC 압축 방법에 대하여 소개하고, 3장에서는 제안하는 고속화 알고리즘을 설명한다. 그리고 4장에서는 실험에 의한 성능분석을 하며 5장에서 결론을 맺는다.

## II. MVC 부호화 방법

서로 다른 시점에서 동일한 장면을 촬영한 세 개 이상의 비디오를 부호화 하는 방법인 MVC의 표준 영상들은 그림 1에서 볼 수 있듯이 보통 8개의 비디오들로 구성되어 있다. MVC를 부호화하는 방법은 상당히 많지만, 기본적으로 다음과 같은 세 가지 종류의 방식을 이용한다. 첫째로 기존의 비디오 압축 방법을 그대로 이용하여 n개의 개별적인 비디오 스트림을 생성하는 방법이다. 이 방법은 가장 직관적이고, 비디오의 시간적인 상관관계만을 이용하여 추가적인 부호화 과정이 필요없다. 두 번째는 각 비디오 간의 상관관계를 이용하는 방법으로, 이웃하고 있는 다른 비디오의 동 시점에 해당하는 장면을 참조하여 부호화 하는 방법이다. 마지막으로 기존의 시간적인 상관관계와 각 시점간의 상관관계를 함께 이용하는 방법으로 최근의 거의 모든 MVC 부호화 과정에서 이용하는 방법이다. 비디오 압축에서 시간적인 상관관계는 움직임 예측(motion estimation)을 통해 적용되는데, MVC에서 각 시점간의 상관관계는 부등 예측(disparity estimation)을 이용하여 적용하게 된다. 전형적인 비디오 부호화에서 사용하는 움직임 예측은 부호화 하려는 프레임의 특정 매크로 블록에 대해 시간 단위로 전후하고 있는 프레임에서 현재 매크로 블록과 가장 유사한 블록의 위치를 벡터형태로 계산하여 부호화 효율을 높이는 방

법이다. 부동 예측은 움직임 예측과 동일한 형태로 구해지게 되는데, 시점이 같고 시간 단위로 전후하고 있는 프레임이 아닌, 이웃하고 있는 다른 시점의 프레임에서 매크로 블록 단위로 부동 벡터를 계산하게 된다. MVC 부호화에서는 움직임 벡터와 부동 벡터를 동시에 이용하여 부호화 효율이 높은 벡터가 선택되게 된다.

일반적인 비디오에서의 마찬가지로 MVC에서도 GOP 단위로 압축을 하게 된다. 이 과정에서 몇 가지 다른 방법들이 제안되었는데, 그 중 대표적인 것이 Mueller의 체계적 B Picture를 이용한 것이다<sup>3)</sup>. 이 방법은 그림 2과 같은 형태로 예측을 하게 되는데, 가로 방향은 시간축 T이며, 세로방향은 시점 즉 개별 스트림을 나타낸 S이다. 그림에서 화살표의 시작점이 화살표의 끝점에 해당하는 프레임이 참조하게 될 프레임을 나타내는데, 한 GOP 단위의 시작과 끝을 이루고 있는 I 프레임에서 예측이 시작된다. 프레임에 적힌 영문자는 각 프레임의 종류인 I, P, B를 나타내며 숫자는 해당 프레임의 참조 레벨을 나타낸다. 영어 대문자의 경우는 피참조 프레임을 의미하며, 소문자일 경우 피참조되지 않는 프레임임을 나타낸다. 또한 숫자가 클 수록 나중에 구성될 레벨이며, 이 레벨을 이용하여 시간적 혹은 공간적인 확장식(scalable) 복호화가 가능하게 된다.

그림에서 볼 수 있듯이 I 프레임에서 먼 프레임부터 예측하기 시작해서 가까운 프레임들까지 체계적으로 예측을 진행하면 시간적인 순서대로 예측할 경우보다 좋은 성능을 보인다<sup>3)</sup>. 또한 비트율-왜

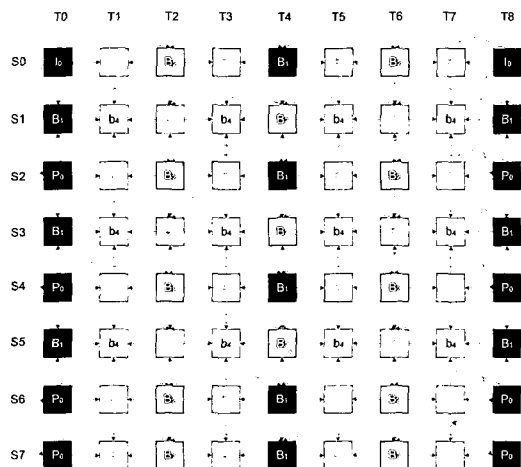


그림 2. 체계적 B picture를 이용한 다시점 비디오 부호화의 구조

Fig 2. The structure of multi-view video encoding using hierarchical B picture

표 1. 총 200프레임의 ballroom 시퀀스를 부호화할 때 각 매크로 블록을 예측하기 위해 필요한 계산 시간

Table 1. Processing time needed to predict each macroblock when encode the ballroom sequence of total 200 frames

|                              | Time(sec)  |
|------------------------------|------------|
| Total MB estimation          | 14,486,375 |
| Intra prediction (I, B, P)   | 124,419    |
| Inter prediction(B1, P)      | 159,298    |
| Inter prediction(B2, B3, b4) | 14,182,040 |
| etc                          | 20,618     |

곡 최적화(rate-distortion optimization)를 이용하여 동일 비트율에서 화질을 높이기 위해서 각 프레임의 양자화 파라미터(Quantization Parameter, QP) 값은 다르게 주어진다<sup>4)</sup>. 또 다른 MVC 방법인 Lattices-like pyramid GOP나 Shared key frame도 방법에 있어서 약간의 다른 점이 있지만 거의 비슷한 형태로 예측이 진행되며, 성능 또한 비슷하다<sup>2)5)6)</sup>.

이와같이 MVC는 기존의 비디오 부호화 방법에 비해 하나의 GOP 내에서 B 프레임의 숫자가 많으며, 다중 참조를 하지 않는다고 가정할 경우 B 프레임에 대한 참조 프레임의 숫자가 두 배로 되기 때문에 인터 예측을 위한 계산량이 많아진다.

그림 2의 I 프레임의 경우에는 인트라 예측만 하면 되지만, P 프레임의 경우에는 인트라 예측에 더해 움직임 벡터 또는 부동 벡터 예측을 해야 하고, B 프레임의 경우에는 인트라 예측과 움직임 벡터, 부동 벡터 예측을 모두 해야 하기 때문에 계산량은 몇 배가량 증가하게 된다. 표 1을 보면 다시점 비디오의 표준 비디오인 ballroom의 I, B, P 프레임을 압축하기 위해 필요한 계산 시간이 나와 있다. 이 수치는 200프레임을 시뮬레이션한 결과이며, 이 중 I 프레임은 2장, B 프레임은 142장이며 P 프레임은 56장이다. 프레임의 개수에서 차이가 난다는 점을 고려하더라도 대부분의 계산이 B프레임의 인터예측에 집중되어 있는 것을 알 수 있다. 표에서 알 수 있듯이 그림 2의 B1이나 P 프레임은 두 개의 참조영상을 이용하지만 B2, B3, b4 프레임의 경우 네 개의 참조영상을 이용하기 때문에 많은 계산시간이 필요한 것을 알 수가 있다. 또한 인트라 예측은 참조 프레임을 사용하지 않으며 움직임 벡터를 계산할 필요가 없으므로 계산시간이 짧은 반면 인터 예측은 많은 계산이 필요함을 알 수 있다.

### III. 제안하는 고속화 알고리즘

앞에서 살펴본 것처럼 H.264/AVC를 이용하여 MVC 부호화를 하는 경우, 부호화에 필요한 대부분의 계산이 인터 예측을 하는 과정에 집중되어 있다. 따라서 효과적으로 부호화 시간을 단축하기 위해서는 인터 예측 부분에서 고속화를 해야 한다. 본 논문에서는 두 가지 고속화 알고리즘을 제시하는데, 하나는 각 GOP 단위의 B 프레임에서 매크로 블록의 모드별 인터 예측을 하는 경우 예측의 왜곡값에 대한 문턱값을 적응적으로 계산하여 경우에 따라 부등 예측을 생략하는 방법이고, 추가적인 방법으로 인터 예측에서 참조된 블록이 시간적으로 이웃한 프레임의 것일 경우 인트라 예측 자체를 생략하는 방법이다.

#### 3.1 적응적 인터-부등 예측의 생략(알고리즘 1)

그림 2에서 살펴본 것처럼 MVC에서 B 프레임은 시간적으로 전후하는 두 장의 프레임으로부터 움직임 예측을 하고, 시점 상으로 좌우에 위치하는 두 장의 프레임으로부터 부등 예측을 해서 총 네 장의 이웃한 프레임으로부터 예측을 하게 된다. 단일 비디오의 H.264/AVC 부호화 방법에서도 다중 프레임으로부터 예측을 사용했으나, 그 경우에는 시간상으로 부호화 하려는 프레임의 전과 후를 이루고 있는 프레임으로부터 예측을 하는 방법으로 [7]에서 효과적으로 이를 고속화 하는 알고리즘을 개발했다. 그러나 MVC에서는 예측에 사용하는 네 장의 프레임이 모두 다른 특성을 가지고 있기 때문에 이런 고속화 방법을 사용할 수 없다.

MVC의 B 프레임에서 사용하는 예측은 인터-움직임, 인터-부등, 인트라의 세 가지로 B 프레임에서 각각의 모드가 차지하는 비율을 측정하면 그림 3과 같다. 마지막으로 1/4 영상에서 계산한 중복 영역의 길이를 이용하여, 원 영상의 중복 영역을 가장 정밀한 정도로 계산하게 된다.

이 모드들 중 인터-움직임의 특별한 형태인 SKIP 예측의 부호화 효율이 가장 좋으며, 인터 예측의 부호화 효율은 인트라 예측에 우선한다. 즉, 시간적으로 전후하는 프레임에서 예측한 인터 블록이 더 일반적이며, 시점 상으로 좌우에 위치하는 프레임에서 예측한 인터 블록은 차순이 된다.

본 논문에서 제안하는 방법은 이런 특징을 이용한 것으로, MVC의 B 프레임의 부호화 과정에서 인터 예측을 인터-움직임, 인터-부등 순으로 진행시

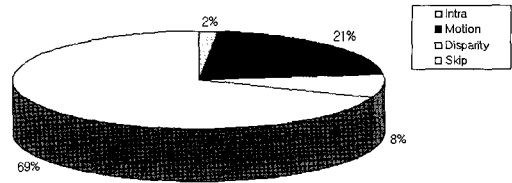


그림 3. 다시점 비디오 부호화시 B 프레임 예측 모드의 분포  
Fig 3. Distribution of estimation modes for a B frame of multi-view video encoding

키고 만일 인터-움직임에서 충분한 부호화 효율을 얻었다면 인터-부등 과정을 생략하는 간단한 알고리즘이다. 이때 인터-움직임 예측에서 계산된 움직임 벡터를 이용해서 매크로 블록을 부호화한다고 가정했을 때 발생하는 왜곡값을 살펴보고 미리 정해진 문턱값(threshold)보다 작은 왜곡을 가질 경우 인터-부등 예측을 생략하게 된다. 문턱값은 매 GOP마다 적응적으로 계산하게 되는데, 해당 GOP의 첫 번째 B 프레임에서 최종적으로 인터-움직임으로 결정된 블록의 왜곡값을 누적하여 계산하게 된다. 매크로 블록은 16x16, 16x8, 8x16, 그리고 8x8의 크기로 결정되기 때문에 이들의 문턱값은 서로 다르게 된다. 따라서 각 블록의 크기에 따라 다르게 누적해서 네 개의 서로 다른 문턱값을 계산하게 되며, 각 문턱값은 해당 크기의 인터 예측을 수행할 때 사용한다.

문턱값은 해당 크기를 가진 블록들의 왜곡 평균을 이용할 수도 있지만, 실험결과에서 특정 프레임의 경우 왜곡값이 넓은 범위를 가지고 분포하기 때문에 다음 식(1)과 같이 왜곡값의 표준편차를 이용하여 문턱값을 계산했다.

$$Threshold = M_{distortion} + \sigma_{add} \quad (1)$$

위 식에서  $M_{distortion}$ 은 왜곡의 평균값이고,  $\sigma_{add}$ 는 왜곡의 표준편차( $\sigma_{distortion}$ )에서 계산된 값이다. 위와 같이 평균에 표준편차의 값을 더해서 평균보다 약간 높은 값으로 문턱값을 결정하여 잘못 예측하는 경우를 줄였으며,  $\sigma_{add}$ 는 표준편차의 값이 너무 큰 경우를 위해 다음과 같이 계산한다.

$$\sigma_{add} = \begin{cases} \text{if } \sigma_{distortion} < M_{distortion} * 0.5 \\ \sigma_{distortion} \times 0.5, \\ \text{else} \\ M_{distortion} \times 0.25 \end{cases} \quad (2)$$

3.2 적응적 인트라 예측의 생략(알고리즘 2)

MVC에서 B 프레임의 매크로 블록은 예측과정을 거쳐서 인트라 블록, 인터-움직임 블록, 인터-부등 블록의 세 가지 중 하나로 결정된다. 그림 3에서 볼 수 있듯이 B 프레임의 매크로 블록은 SKIP, 인터-움직임, 또는 인터-부등 블록으로 예측된다. 실제로 인트라로 예측되는 경우는 전체 매크로 블록의 2%정도로 상당히 적은 수임을 알 수 있는데, 이는 대부분의 매크로 블록을 예측할 때 인트라 예측을 사용하지 않아도 된다는 것이다. 두 번째 고속화 방법은 이 점을 이용한 것으로 인트라에 앞서 인터 예측을 수행하고, 인터 예측에서 결정된 블록이 인터-움직임 블록이거나 인터-부등 블록이면서 미리 정해진 문턱값보다 작은 왜곡값으로 예측된 블록이면 인트라 예측을 생략하는 방법이다. 따라서 이 방법은 알고리즘 1과 연결하여 사용할 수 있으며, 그림 4에 알고리즘 1에 알고리즘 2를 추가한 블록 다이어그램을 보이고 있다.

그림 4를 보면 B 프레임을 구성하는 매크로 블록 단위로 알고리즘이 진행되는 것을 알 수 있는데, 인터-움직임 예측은 어떤 매크로 블록이든지 실행하게 된다. 만일 매크로 블록이 속해있는 프레임이 해당 GOP의 첫 번째 B 프레임이라면 문턱값을 계산하기 위해 인터-움직임 예측에서 계산된 왜곡값을

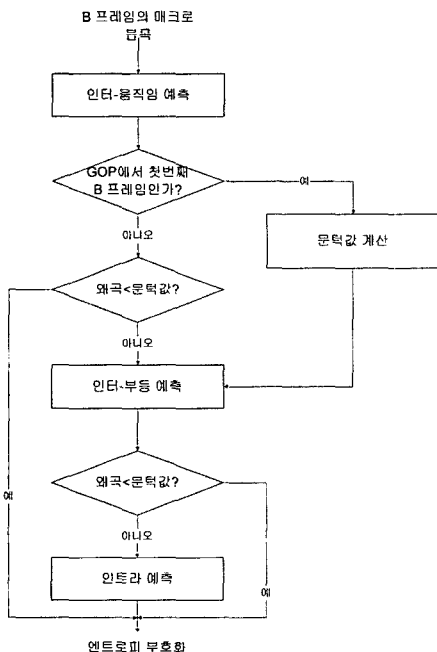


그림 4. 제안하는 알고리즘의 블록 다이어그램  
Fig 4. A block diagram of proposed algorithm

표 2. 제안하는 방법으로 부호화 했을 경우 기존의 방법에 대해 발생하는 예측 오류의 예

Table 2. An example of prediction error for encoded result using proposed algorithm compared to existing method

| Failed prediction  | # blocks | Proportion to total blocks(%) |
|--------------------|----------|-------------------------------|
| 16x16              | 3,412    | 20.0                          |
| 16x8               | 759      | 4.45                          |
| 8x16               | 1,250    | 7.34                          |
| 8x8                | 1,148    | 6.74                          |
| Intra16            | 162      | 0.95                          |
| Intra8             | 46       | 0.27                          |
| Intra4             | 6        | 0.04                          |
| for 170,400 blocks |          |                               |

누적하게 되는데, 실제로는 매크로 블록의 각 모드(16x16, 16x8, 8x16, 8x8) 별로 누적된다. 만일 매크로 블록이 속한 프레임이 첫 번째 B 프레임이 아니라면 인터-움직임의 왜곡값과 앞서 계산된 문턱값을 비교하여 알고리즘 1에서 처럼 인터-부등 예측을 생략하게 된다. 또한 이 경우 알고리즘 2에서 인트라 예측을 생략하기 때문에 해당 매크로 블록은 인터-움직임 예측으로 부호화하게 된다. 만일 문턱값보다 왜곡이 크면 움직임-부등 예측을 하게 되고, 움직임-부등에서 계산된 왜곡값을 알고리즘 1에서 사용했던 문턱값과 비교하여 인트라 예측 수행여부를 결정한다.

표 2에 Ballroom 실험 비디오 142프레임(170,400개의 매크로 블록)을 JSVM 참조 소프트웨어를 이용하여 부호화했을 때와 제안하는 방법으로 부호화한 경우, 두 결과의 차이점을 천분율(permill) 단위로 나타냈다. 이 표에서 알 수 있듯이 인터-움직임 예측을 한 결과의 왜곡값이 미리 정해진 문턱값보다 작은 경우, 인터-부등 블록의 왜곡값이 더 작아지는 빈도는 상당히 적다는 것을 알 수 있다. 다음장에서 자세한 실험 결과 및 제안한 방법의 성능을 알아보도록 하겠다.

IV. 실험 결과

이 장에서는 제안하는 알고리즘을 이용한 실험 과정과 결과를 제시한다.

성능을 평가하기 위해 6종류의 표 3과 같은 ISO/IEC JTC 1/SC 29/WG11 테스트 영상을 사용했다<sup>[8][9]</sup>. 표에서 보는 것처럼 실험에 사용한 비디

오는 실제 방송용으로 적합하도록 해상도가 상당히 높은 것을 알 수 있다. 그리고 각 비디오는 해상도, 촬영한 시점의 개수, 전역 움직임(global motion)의 유무 등이 각기 다르게 되어있어서, 여러 조건에서 성능을 평가할 수 있도록 되어있다.

고속화 알고리즘은 ISO/IEC JTC 1/SC 29/WG11의 참조 소프트웨어인 JSVM3.5에 구현했으며, 3.6GHz로 동작하는 Intel Pentium 4 CPU가 장착된 컴퓨터를 실험에 사용했다<sup>[10]</sup>. 또한 현재 표준화에서 비트율-왜곡 성능이 좋다고 인정된 방법을 사용하기 위해 GOP 내의 각 프레임의 QP를 레벨에 따라 다르게 설정하였다. 또한 GOP는 일반적으로 0.5초 단위로 설정되므로, 실험 영상의 프레임율에 따라 시점당 12또는 15프레임을 한 GOP로 설정하였다. 단, Breakdancers는 프레임율이 15fps 이지만 15프레임을 한 GOP로 설정하였다.

실험에서 설정한 파라미터는 표 4와 같으며, 참조 소프트웨어와 제안한 방법 모두 동일한 조건에서 실험이 이루어졌다. 표 4에서 differential QP는 부호화 효율을 높이기 위해 사용되는데, 앞선 그림 2에서의 I, P 프레임은 원래의 QP값으로 부호화 하고, B1, B2, B3, b4는 각각 해당하는 값을 더해서 QP를 설정할 때 사용된다.

실험은 부호화 시작 시점부터 모든 파일을 생성

표 3. 실험에 사용된 표준 비디오 시퀀스  
Table 3. Standard video sequences for the experiments

| Name         | Resolution | # View | GOP Length | #used Frames |
|--------------|------------|--------|------------|--------------|
| Ballroom     | 640×480    | 8      | 12         | 200          |
| Exit         | 640×480    | 8      | 12         | 200          |
| Flamenco2    | 640×480    | 5      | 15         | 155          |
| Race1        | 640×480    | 8      | 15         | 248          |
| Breakdancers | 1024×768   | 8      | 15         | 248          |
| Uli          | 1024×768   | 8      | 12         | 200          |

표 4. 실험에 사용된 설정 파라미터  
Table 4. Configuration parameters for the experiments

| Feature                   | Parameter     |
|---------------------------|---------------|
| Symbol mode               | CABAC         |
| differential QP           | 0, 3, 4, 5, 6 |
| Search mode               | fast search   |
| Search function(Full-pel) | SAD-YUV       |
| Search function(Sub-pel)  | HADAMARD      |
| Search Range              | 96            |
| Loop Filter               | ON            |
| RD Optimization           | ON            |

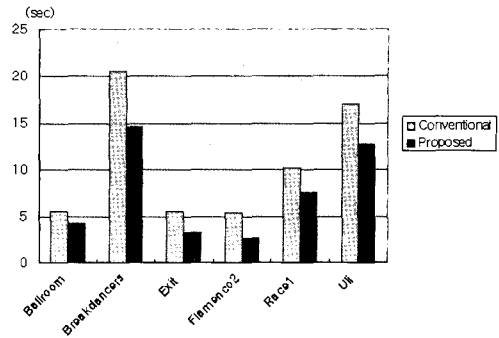


그림 5. 다양한 표준 비디오에 대한 기존의 방법과 제안한 방법의 부호화 시간 비교  
Fig 5. Encoding time comparison of existing method and proposed method for various standard videos

표 5. 다양한 표준 비디오에 대한 기존의 방법과 제안한 방법의 정량적 성능 비교  
Table 5. Encoding time quantitative comparison of existing method and proposed method for various standard videos

| Name         | QP | Conventional JSVM (sec) | Proposed Algorithm (sec) | Improvement (%) |
|--------------|----|-------------------------|--------------------------|-----------------|
| Ballroom     | 29 | 5,471                   | 4,197                    | 23.3            |
| Exit         | 26 | 5,470                   | 3,311                    | 39.5            |
| Flamenco2    | 28 | 5,425                   | 2,687                    | 50.5            |
| Race1        | 24 | 10,169                  | 7,531                    | 25.9            |
| Breakdancers | 22 | 20,598                  | 14,585                   | 29.2            |
| Uli          | 28 | 17,073                  | 12,786                   | 25.1            |

하고 종료할 때까지의 시간을 측정했는데, 결과의 성능 비교는 그림 5의 그래프를 통해 볼 수 있다.

표 5는 실험에 사용된 비디오의 QP와, 정량적인 비교를 위한 실험수치를 표시했다. 모든 표준 비디오에 대해 평균 32%의 속도 개선이 있었으며, 특히 Flamenco2 비디오의 경우는 50% 이상의 속도 개선이 있었다. Flamenco2의 개선 정도가 다른 표준 비디오에 비해 높은 이유는 다른 표준 비디오와는 다르게 Flameco2는 십자형으로 배열된 카메라와 중앙의 카메라에서 비디오를 획득하여, 중앙의 비디오가 다른 모든 비디오의 참조가 되면서 인터-부동 예측을 하는 B 프레임의 발생빈도가 높아졌기 때문으로 판단된다.

그림 6에는 기존의 방법과 제안하는 방법으로 부호화한 결과의 비트율-왜곡을 비교했는데, 그림 상으로 구분하기 어려울 정도로 큰 차이를 보이지 않음을 알 수 있다.

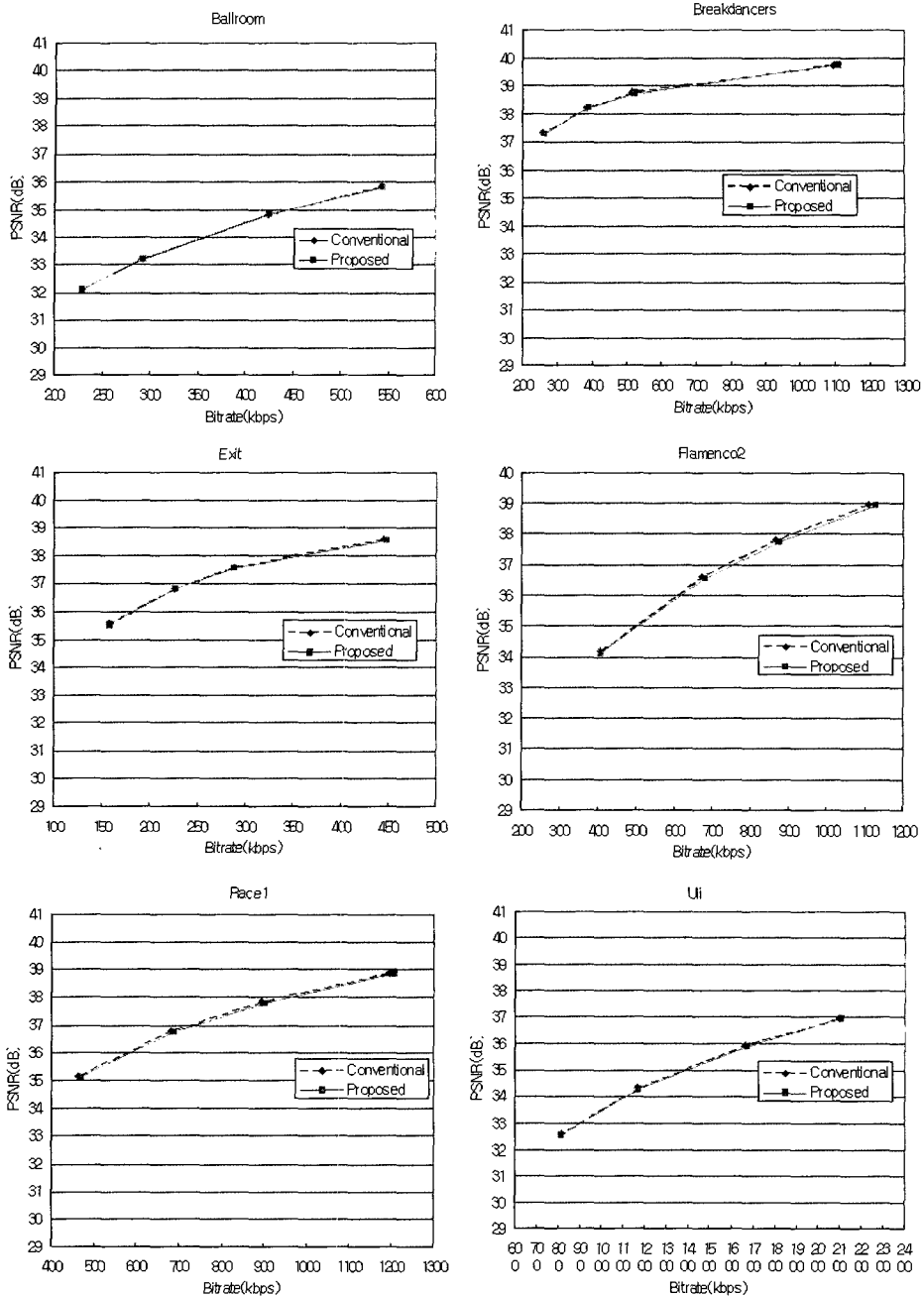


그림 6. 기존의 방법에 대한 제안한 방법의 각 실험별 비트율-왜곡 곡선  
 Fig. 6. A rate-distortion curve of each experiment for proposed method compared to existing method

### V. 결론

H.264/AVC는 월등한 부호화 성능을 가졌지만 복잡도가 크다는 단점이 있으며, 동시에 여러 비디오를 전송해야 하는 다시점 비디오를 부호화할 경우 그 계산량은 엄청나게 크다. 본 논문에서는 GOP

단위로 적응적인 문턱값을 설정하여 인터-부동 및 인트라 예측을 생략하는 H.264/AVC 다시점 비디오의 고속 부호화 방법에 대해 제안했다. 앞선 실험 결과로 제안하는 방법이 기존 방법에 비해 부호화 시간을 상당히 단축할 수 있다는 것을 증명하였다. 또한 고속화 알고리즘의 단점인 화질의 저하 및 비

트율의 증가도 상당히 적어서 원래 부호화 방법에 비해 손색이 없다는 것도 살펴보았다. 실험에서는 두 개의 GOP 단위를 부호화 했지만 더 많은 프레임을 부호화 할 경우 좀 더 많은 이점을 볼 수 있으며, 제안하는 방법이 B 프레임에 관련된 방법이므로 GOP의 길이가 길어질수록 고속화 성능은 더 좋아질 것이다. 또한 P 프레임의 고속화 알고리즘과 인터라 예측의 고속화 알고리즘 등과 함께 사용하면 다시점 비디오 부호화 시간을 상당히 줄일 수 있을 것이다.

### 참 고 문 헌

- [ 1 ] A. Smolic and P. Kauff, "Interactive 3D Video Representation and Coding Technologies," Proceedings of the IEEE, Special Issue on Advances in Video Codings and Delivery, Vol.93, no.1, pp.99-110, Jan. 2005.
- [ 2 ] Kwan-Jung Oh and Yo-Sung Ho, "Multi-view Video Coding based on the Lattices-like Pyramid GOP Structure," Picture Coding Symposium (PCM2006), p2-18, Apr. 2006.
- [ 3 ] K. Mueller, P. Merkle et al., "Multi-view Video Coding based on H.264/MPEG4-AVC using Hierarchical B Pictures," Picture Coding Symposium (PCM2006), SS3-3, Apr. 2006.
- [ 4 ] H. Schwartz, D. Marpe and T. Wiegand, "Hierarchical B pictures," Join Video Team, JVT-P014, July 2005.
- [ 5 ] X. Cheng, L. Sun and S. Yang, "A Multi-view video coding scheme using Shared Key Frames for High Interactive Application," Picture Coding Symposium (PCM2006), p2-21, Apr. 2006.
- [ 6 ] ISO/IEC JTC1/SC29/WG11, "Survey of Algorithms used for Multi-view Video Coding(MVC)," N6909, Jan. 2005.
- [ 7 ] W. LEE, J. LEE, I. Cho and D. Jeong, "Reduction of Mode Decision Complexity in H.264/AVC using Adaptive Selection of Reference Frame and Intra Prediction Mode," ICCSA 2006 (LNCS 3980), pp.326-334, May 2006.
- [ 8 ] ISO/IEC JTC1/SC29/WG11, "Multiview Video Test Sequences from MERL," m12077, Apr. 2005.
- [ 9 ] ISO/IEC JTC1/SC29/WG11, "Call for Proposals on Multi-view Video Coding," n7567, Oct. 2005.
- [ 10 ] Multi-view Coding Reference Software, [http://iphome.hhi.de/mueller/MVC\\_SW.htm](http://iphome.hhi.de/mueller/MVC_SW.htm)

이 정 호 (Jungho Lee)

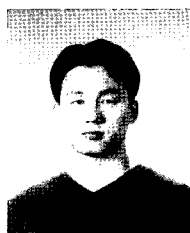
정회원



1999년 2월 인하대학교 전자공학과 공학사  
 2001년 2월 인하대학교 전자공학과 석사  
 2001년 3월~현재 인하대학교 전자공학과 박사과정  
 <관심분야> Multi-view Video Coding, Scalable Video Coding

조 익 환 (Ikhwan Cho)

정회원



2000년 2월 인하대학교 전자공학과 졸업  
 2002년 2월 인하대학교 전자공학과 공학석사  
 2002년 3월~현재 인하대학교 전자공학과 박사과정  
 <관심분야> 영상처리, MPEG-4, MPEG-7, SVC(Scalable Video Coding), 의료영상 처리, 이미지 워터마킹

이 웅 호 (Woongho Lee)

정회원



1995년 2월 인하대학교 전자공학과 공학사  
 2002년 2월 인하대학교 전자공학과 공학석사  
 2006년 8월 인하대학교 전자공학과 공학박사  
 2006년 9월~현재 LG전자 DTV 연구소 선임연구원  
 <관심분야> 영상처리, Video Coding, Video and Image Segmentation

정 동 석 (Dongseok Jeong)

정회원



1977년 2월 서울대학교 전기공학 졸업  
 1985년 Virginia 주립 공과대학 전자공학과 공학석사  
 1988년 Virginia 주립 공과대학 전자공학과 공학박사  
 1988년 3월~현재 인하대학교 전자공학과 교수  
 <관심분야> 영상처리, 컴퓨터 비전, 멀티미디어 정보 처리